

AMENDMENTS TO THE CLAIMS

This claims lists supersedes all previous claims lists.

WHAT IS CLAIMED IS

1. (Cancelled)
2. (Currently Amended) A method for reusing a data structure in which a compiled cursor is stored, comprising:
receiving a database statement from a client;
analyzing the statement;
determining if the database statement is optimally shareable, sub-optimally shareable, or non-shareable based on the analysis;
searching server memory for a similar database statement; and
reusing the data structure compiled for the similar database statement to execute the database statement when a system parameter that is configurable to control data structure sharing ~~The method of claim 1, wherein reusing said data structure compiled for said similar database statement to execute said database statement occurs if said system parameter is configured to enable data structure sharing for all shareable database statements.~~
3. (Currently Amended) A method for reusing a data structure in which a compiled cursor is stored, comprising:
receiving a database statement from a client;
analyzing the database statement;
determining if the database statement is optimally shareable, sub-optimally shareable, or non-shareable based on the analysis;
searching server memory for a similar database statement; and
reusing the data structure compiled for the similar database statement to execute the database statement when a system parameter that is configurable to control data structure sharing ~~The method of claim 1, wherein reusing said data structure compiled for~~

~~said similar database statement to execute said database statement occurs if said system~~
parameter is configured to enable data structure sharing only for optimally shareable
database statements.

4. (Currently Amended) The method of claim 12, wherein reusing said data
structure compiled for said similar database statement occurs when a command-line
parameter is configured to control data structure sharing.

5. (Original) The method of claim 4, wherein the command-line parameter is a hint
for enabling a one-time system parameter override.

6. (Currently Amended) A method for reusing a data structure in which a compiled
cursor is stored, comprising:

receiving a database statement from a client;

analyzing the database statement;

determining if the database statement is optimally shareable, sub-optimally
shareable, or non-shareable based on the analysis;

~~The method of claim 1, further comprising:~~

searching server memory for an exact matching database statement ~~prior to said~~
~~step of searching memory for said similar database statement~~ if the statement is non-
shareable; and

executing the data structure compiled from said exact matching database
statement if said exact matching database statement is found in said memory.

7. (Currently Amended) The method of claim 12, further comprising:

replacing all literals in said database statement with bind variables prior to said
step of searching memory for said similar database statement.

8. (Currently Amended) The method of claim 12, further comprising returning a
result set to the client if said database statement is a query.

9. (Currently Amended) The method of claim 12, wherein reusing said data structure includes executing the execution plan compiled from said similar database statement.
10. (Currently Amended) The method of claim 12, ~~wherein said data structure comprises a cursor~~ wherein analyzing the database statement comprising analyzing one or more of the group of: a position of a literal, a value of the literal, a number of rows in a table, and the type of database statement.
11. - 44. (Cancelled)
45. (Currently Amended) A system for sharing a data structure in which a compiled cursor is stored, comprising:
a server configured to accept a database statement from a client, analyze the statement, and to determine if the database statement is optimally shareable, sub-optimally shareable, or non-shareable based on the analysis;
a search engine for searching shared memory for a similar database statement; and
a cursor sharing monitor for executing a data structure compiled for said similar database statement if the value of a system parameter is configured to enable data structure sharing ~~The system of claim 44,~~ wherein the cursor sharing monitor reuses an execution plan of the data structure compiled for the similar database statement to execute the database statement only if the system parameter is configured to enable data structure sharing for all shareable database statements.
46. (Currently Amended) A system for sharing a data structure in which a compiled cursor is stored, comprising:
a server configured to accept a database statement from a client, analyze the statement, and to determine if the database statement is optimally shareable, sub-optimally shareable, or non-shareable based on the analysis;
a search engine for searching shared memory for a similar database statement; and

a cursor sharing monitor for executing a data structure compiled for said similar database statement if the value of a system parameter is configured to enable data structure sharing The system of ~~claim 44~~, wherein the cursor sharing monitor reuses an execution plan of the data structure compiled for the similar database statement to execute the database statement only if said system parameter is configured to enable data structure sharing for all optimally shareable database statements.

47. (Cancelled)

48. (Currently Amended) The system of claim 45 ~~claims 44~~, further comprising a compiler for hard parsing said database statement.

49. (Original) The system of claim 48, wherein the compiler hard parses the database statement only if (1) the value of said system parameter is not configured to force data structure sharing and said database statement and said similar database statement are sub-optimally similar, or (2) if said system parameter is configured for data structure sharing only if said database statement exactly matches said similar database statement and said database statement is optimally shareable with said similar database statement, or (3) said database statement is non-shareable with said similar database statement.

50. (Original) The system of claim 48, wherein the compiler further comprises an optimizer for creating an optimal execution plan for carrying out the database statement from the client.

51. (Original) The system of claim 49, wherein the compiler further comprises a typechecker for resolving data type conflicts between the server and the client.

52. (Original) The system of claim 49, wherein the compiler further comprises a parser for hard parsing the database statement.

53. (Original) The system of claim 52, wherein the parser further comprises a tree builder for building an expression tree from the database statement.

54. (Original) The system of claim 52, wherein the parser further comprises a semantic analyzer for verifying user permissions and access privileges for one or more objects referenced in the database statement.
55. (Original) The system of claim 49, wherein the parser further comprises a syntactic analyzer for checking the syntax of the database statement.
56. - 68. (Cancelled)
69. (New) The method of claim 3, wherein reusing said data structure compiled for said similar database statement occurs when a command-line parameter is configured to control data structure sharing.
70. (New) The method of claim 69, wherein the command-line parameter is a hint for enabling a one-time system parameter override.
71. (New) The method of claim 3, further comprising:
replacing all literals in said database statement with bind variables prior to said step of searching memory for said similar database statement.
72. (New) The method of claim 3, further comprising returning a result set to the client if said database statement is a query.
73. (New) The method of claim 3, wherein reusing said data structure includes executing the execution plan compiled from said similar database statement.
74. (New) The method of claim 3, wherein analyzing the database statement comprising analyzing one or more of the group of: a position of a literal, a value of the literal, a number of rows in a table, and the type of database statement.
75. (New) The method of claim 6, wherein reusing said data structure compiled for said similar database statement occurs when a command-line parameter is configured to control data structure sharing.

76. (New) The method of claim 75, wherein the command-line parameter is a hint for enabling a one-time system parameter override.
77. (New) The method of claim 6, further comprising:
replacing all literals in said database statement with bind variables prior to said step of searching memory for said similar database statement.
78. (New) The method of claim 6, further comprising returning a result set to the client if said database statement is a query.
79. (New) The method of claim 6, wherein reusing said data structure includes executing the execution plan compiled from said similar database statement.
80. (New) The method of claim 6, wherein said data structure comprises a cursor.
81. (New) A computer program product embodied on computer readable medium, the computer readable medium having stored thereon a sequence of instructions which, when executed by a processor, causes the processor to execute a method for reusing a data structure in which a compiled cursor is stored, the method comprising:
receiving a database statement from a client;
analyzing the statement;
determining if the database statement is optimally shareable, sub-optimally shareable, or non-shareable based on the analysis;
searching server memory for a similar database statement; and
reusing the data structure compiled for the similar database statement to execute the database statement when a system parameter that is configurable to control data structure sharing is configured to enable data structure sharing for all shareable database statements.
82. (New) The computer program product of claim 81, wherein reusing said data structure compiled for said similar database statement occurs when a command-line parameter is configured to control data structure sharing.

83. (New) The computer program product of claim 82, wherein the command-line parameter is a hint for enabling a one-time system parameter override.

84. (New) The computer program product of claim 81, further comprising:
replacing all literals in said database statement with bind variables prior to said step of searching memory for said similar database statement.

85. (New) The computer program product of claim 81, further comprising returning a result set to the client if said database statement is a query.

86. (New) The computer program product of claim 81, wherein reusing said data structure includes executing the execution plan compiled from said similar database statement.

87. (New) The computer program product of claim 81, wherein analyzing the database statement comprising analyzing one or more of the group of: a position of a literal, a value of the literal, a number of rows in a table, and the type of database statement.

88. (New) A computer program product embodied on computer readable medium, the computer readable medium having stored thereon a sequence of instructions which, when executed by a processor, causes the processor to execute a method for reusing a data structure in which a compiled cursor is stored, the method comprising:

receiving a database statement from a client;

analyzing the database statement;

determining if the database statement is optimally shareable, sub-optimally shareable, or non-shareable based on the analysis;

searching server memory for a similar database statement; and

reusing the data structure compiled for the similar database statement to execute the database statement when a system parameter that is configurable to control data structure sharing is configured to enable data structure sharing only for optimally shareable database statements.

89. (New) The computer program product of claim 88, wherein reusing said data structure compiled for said similar database statement occurs when a command-line parameter is configured to control data structure sharing.
90. (New) The computer program product of claim 89, wherein the command-line parameter is a hint for enabling a one-time system parameter override.
91. (New) The computer program product of claim 88, further comprising:
replacing all literals in said database statement with bind variables prior to said step of searching memory for said similar database statement.
92. (New) The computer program product of claim 88, further comprising returning a result set to the client if said database statement is a query.
93. (New) The computer program product of claim 88, wherein reusing said data structure includes executing the execution plan compiled from said similar database statement.
94. (New) The computer program product of claim 88, wherein analyzing the database statement comprising analyzing one or more of the group of: a position of a literal, a value of the literal, a number of rows in a table, and the type of database statement.
95. (New) A computer program product embodied on computer readable medium, the computer readable medium having stored thereon a sequence of instructions which, when executed by a processor, causes the processor to execute a method for reusing a data structure in which a compiled cursor is stored, the method comprising:
receiving a database statement from a client;
analyzing the database statement;
determining if the database statement is optimally shareable, sub-optimally shareable, or non-shareable based on the analysis;

searching server memory for an exact matching database statement if the statement is non-shareable; and

executing the data structure compiled from said exact matching database statement if said exact matching database statement is found in said memory.

96. (New) The computer program product of claim 95, wherein reusing said data structure compiled for said similar database statement occurs when a command-line parameter is configured to control data structure sharing.

97. (New) The computer program product of claim 96, wherein the command-line parameter is a hint for enabling a one-time system parameter override.

98. (New) The computer program product of claim 95, further comprising:
replacing all literals in said database statement with bind variables prior to said step of searching memory for said similar database statement.

99. (New) The computer program product of claim 95, further comprising returning a result set to the client if said database statement is a query.

100. (New) The computer program product of claim 95, wherein reusing said data structure includes executing the execution plan compiled from said similar database statement.

101. (New) The computer program product of claim 95, wherein analyzing the database statement comprising analyzing one or more of the group of: a position of a literal, a value of the literal, a number of rows in a table, and the type of database statement.

102. (New) The system of claim 46, further comprising a compiler for hard parsing said database statement.

103. (New) The system of claim 102, wherein the compiler hard parses the database statement only if (1) the value of said system parameter is not configured to force data

structure sharing and said database statement and said similar database statement are sub-optimally similar, or (2) if said system parameter is configured for data structure sharing only if said database statement exactly matches said similar database statement and said database statement is optimally shareable with said similar database statement, or (3) said database statement is non-shareable with said similar database statement.

104. (New) The system of claim 102, wherein the compiler further comprises an optimizer for creating an optimal execution plan for carrying out the database statement from the client.

105. (New) The system of claim 103, wherein the compiler further comprises a typechecker for resolving data type conflicts between the server and the client.

106. (New) The system of claim 103, wherein the compiler further comprises a parser for hard parsing the database statement.

107. (New) The system of claim 106, wherein the parser further comprises a tree builder for building an expression tree from the database statement.

108. (New) The system of claim 106, wherein the parser further comprises a semantic analyzer for verifying user permissions and access privileges for one or more objects referenced in the database statement.

109. (New) The system of claim 103, wherein the parser further comprises a syntactic analyzer for checking the syntax of the database statement.